

CLAIM AMENDMENTS

Claim 1 (Previously Presented): A method of hierarchical scheduling comprising:

receiving, by a network processor, data from one or more pipes, each pipe including a plurality of pipe flows;

selecting, by the network processor, a winning pipe from the one or more pipes from which to transmit data based upon one or more quality of service parameters corresponding to the winning pipe;

selecting, by the network processor, a winning pipe flow from the plurality of pipe flows included in the winning pipe based upon one or more quality of service parameters corresponding to the winning pipe flow; and

transmitting, by the network processor, data from the winning pipe flow using a bandwidth corresponding to the winning pipe flow.

Claim 2 (Canceled)

Claim 3 (Original) The method of claim 1 wherein selecting a winning pipe from the one or more pipes from which to transmit data based upon one or more quality of service parameters corresponding to the winning pipe includes writing data identifying a pipe to a memory address in a group of memory addresses based upon one or more quality of service parameters corresponding to the pipe and scanning the group of memory addresses to find data identifying a pipe.

Claim 4 (Original) The method of claim 3 further comprising rewriting data identifying the winning pipe to a memory address

in a group of memory addresses based upon one or more quality of service parameters corresponding to the winning pipe.

Claim 5 (Previously Presented) The method of claim 1 wherein selecting a winning pipe flow from the plurality of pipe flows included in the winning pipe, based upon one or more quality of service parameters corresponding to the winning pipe flow, includes:

- writing data identifying a pipe flow to a memory address in a group of memory addresses based upon one or more quality of service parameters corresponding to the pipe flow;

- scanning the group of memory addresses to find data identifying a pipe flow;

- writing the identified pipe flow in a queue corresponding to the winning pipe based upon one or more quality of service parameters corresponding to the winning pipe flow; and

- selecting the identified pipe flow from the queue corresponding to the winning pipe.

Claim 6 (Previously Presented) The method of claim 5 further comprising writing data identifying the winning pipe flow to a memory address in a group of memory addresses, based upon one or more quality of service parameters corresponding to the winning pipe flow.

Claim 7 (Previously Presented) A method for hierarchical scheduling comprising:

- receiving, by a network processor, data identifying a pipe flow, the pipe flow included in a pipe;

- writing, by the network processor, data regarding the pipe to a first calendar;

writing, by the network processor, data regarding the pipe flow to a second calendar;

scanning, by the network processor, the first calendar for a winning pipe;

scanning, by the network processor, the second calendar for a winning pipe flow;

writing, by the network processor, the winning pipe flow to a corresponding pipe queue;

using, by the network processor, the winning pipe to select a pipe flow from a corresponding pipe queue; and

transmitting, by the network processor, data from the selected pipe flow.

Claim 8 (Original) The method of claim 7 further comprising rewriting data regarding the winning pipe flow to the second calendar.

Claim 9 (Original) The method of claim 7 further comprising rewriting data regarding the winning pipe to the first calendar.

Claim 10 (Previously Presented) A network processor comprising:

at least one memory adapted to store one or more quality of service parameters corresponding to one or more pipes and pipe flows; and

scheduler logic, coupled to the at least one memory, adapted to:

receive data from one or more pipes, each pipe including a plurality of pipe flows;

select a winning pipe from the one or more pipes from which to transmit data based upon one or more quality of service parameters corresponding to the winning pipe;

select a pipe flow from the plurality of pipe flows included in the winning pipe based upon one or more quality of service parameters corresponding to the selected pipe flow; and
transmit data from the selected pipe flow;
wherein the scheduler logic comprises:
a primary calendar for storing at least one of an autonomous flows and a pipe that are scheduled to be serviced;
a secondary calendar for storing pipe flows that are scheduled to be serviced; and
a pipe queue table for storing a winning pipe flow in a queue for a pipe to which the pipe flow corresponds.

Claim 11 (Canceled)

Claim 12 (Previously Presented) The network processor of claim 10 wherein the scheduler logic further comprises:

enqueue and new attach logic for scheduling at least one of an autonomous flow and a pipe flow to be serviced; and

dequeue and reattach logic for selecting at least one of an autonomous flow and a pipe flow to be serviced.

Claim 13 (Original) The network processor of claim 10 wherein the scheduler logic is further adapted to transmit data from the selected pipe flow using a bandwidth corresponding to the winning pipe flow.

Claim 14 (Original) The network processor of claim 10 wherein the scheduler logic is further adapted to write data identifying a pipe to a memory address in a group of memory addresses based upon one or more quality of service parameters corresponding to

the pipe and scanning the group of memory addresses to find data identifying a pipe.

Claim 15 (Original) The network processor of claim 14 wherein the scheduler logic is further adapted to rewrite data identifying the winning pipe to a memory address in a group of memory addresses based upon one or more quality of service parameters corresponding to the winning pipe.

Claim 16 (Original) The network processor of claim 10 wherein the scheduler logic is further adapted to:

- write data identifying a pipe flow to a memory address in a group of memory addresses based upon one or more quality of service parameters corresponding to the pipe flow;

- scan the group of memory addresses to find data identifying a pipe flow;

- write the identified pipe flow in a queue corresponding to the winning pipe based upon one or more quality of service parameters corresponding to the selected pipe flow; and

- select the identified pipe flow from the queue corresponding to the winning pipe.

Claim 17 (Original) The network processor of claim 16 wherein the scheduler logic is further adapted to write data identifying the selected pipe flow to a memory address in a group of memory addresses, based upon one or more quality of service parameters corresponding to the selected pipe flow.

Claim 18 (Original) A network processor comprising:

- at least one memory adapted to store one or more quality of service parameters corresponding to one or more pipes and pipe flows; and

- scheduler logic comprising a first calendar and a second calendar, coupled to the at least one memory and adapted to:

- receive data identifying a pipe flow, the pipe flow included in a pipe;
 - write data regarding the pipe to the first calendar;
 - write data regarding the pipe flow to the second calendar;
 - scan the first calendar for a winning pipe;
 - scan the second calendar for a winning pipe flow;
 - write the winning pipe flow to a corresponding pipe queue;
 - use the winning pipe to select a pipe flow from a corresponding pipe queue; and
 - transmit data from the selected pipe flow.

Claim 19 (Original) The network processor of claim 18 wherein the scheduler logic further comprises: a pipe queue table for storing a winning pipe flow in a queue for a pipe to which the pipe flow corresponds;

- an enqueue and new attach logic for scheduling at least one of an autonomous flow and a pipe flows to be serviced; and

- a dequeue and reattach logic for selecting at least one of an autonomous flow and a pipe flow to be serviced.

Claim 20 (Original) The network processor of claim 18 wherein the scheduler logic is further adapted to rewrite data regarding the winning pipe flow to the second calendar.

Claim 21 (Original) The network processor of claim 18 wherein the scheduler logic is further adapted to rewrite data regarding the winning pipe to the first calendar.